



第四章 选择结构程序设计

主讲人：文安洁

第四章教学目标

教学目标:

- 掌握关系运算符及关系表达式的使用
- 掌握逻辑运算符及逻辑表达式的使用
- 掌握条件语句和多分支选择语句switch编程应用

教学内容:

- 4.1 关系运算符与关系表达式
- 4.2 逻辑运算符与逻辑表达式
- 4.3 if语句
- 4.4 switch语句
- 4.5 程序设计举例

4.1 关系运算符与关系表达式

4.1.1 关系运算符

比较两个数据大小的运算符称为关系运算符。C语言关系运算符如下：

运算符	含义	优先级
>	大于	高
<	小于	
>=	大于等于	
<=	小于等于	
==	等于	低
!=	不等于	

4.1.2 关系表达式

用关系运算符将两个表达式连接起来，进行关系运算的式子，称为关系表达式。

$(x=8) > (y=7)$

将2个赋值表达式连接的关系表达式

$x+y > y+z$

将2个算术表达式连接的关系表达式

$'a' < 'c'$

将2个字符表达式连接的关系表达式

$(x > y) < (y < z)$

将2个关系表达式连接的关系表达式

关系表达式的值：关系成立用“1”表达逻辑真；关系不成立用“0”表达逻辑假。

如：判断数的奇偶性： $i \% 2 \neq 0$ ，若关系成立，表明i是奇数；若关系不成立，表明i是偶数。

4.2 逻辑运算符与逻辑表达式

4.2.1 逻辑运算符

逻辑运算符

与(&&) : 同真为真, 其余为假

或(||) : 同假为假, 其余为真

非(!) : 非真即假, 非假即真

逻辑运算符优先级顺序: 非! > 与&& > 或||

4.2.2 逻辑表达式

通过逻辑运算符将一个或多个表达式连接起来，构成的符合C语言规则的式子称为逻辑表达式。

$(a < (b + c)) \&\& (b < (a + c)) \&\& (c < (a + b))$

该表达式实现对任意输入三角形的三条边a,b,c判断能否构成三解形。

能：表明表达式值为真用“1”表示；

不能：表明表达式值为假用“0”表示。

特征：多个**与** (**&&**) 运算符连接时，左边出现表达式为0，则停止运算；
多个**或** (**||**) 运算符连接时，左边出现表达式为1，则停止运算。

关系运算符逻辑运算符的应用

1. 字符型变量ch是英文字母的表示式。

```
(ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z')
```

2. 字符型变量ch是数字的表示式。

```
ch >= '0' && ch <= '9'
```

3. 整型变量x、y不能同时为0的表达式。

```
x != 0 || y != 0
```

4. 判断整型变量year是闰年的表达式。

(提示: year能被4整除但不能被100整除或year能被400整除)

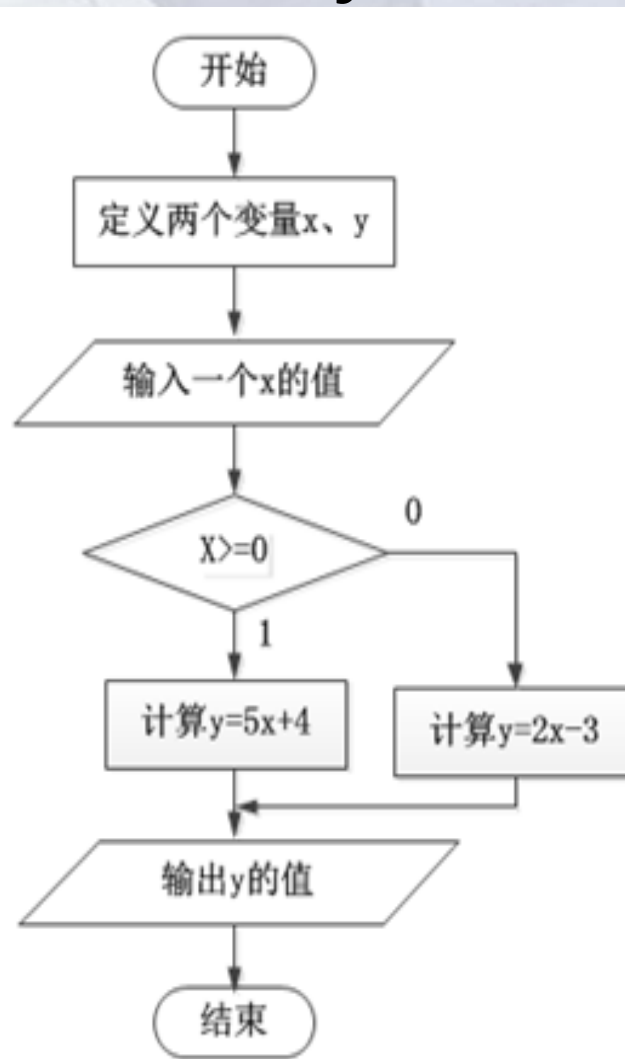
```
(year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)
```

4.3 if语句

问题：计算分段函数的值，根据输入x的值输出函数y的值。

$$y = \begin{cases} 5x+4 & (x \geq 0) \\ 2x-3 & (x < 0) \end{cases}$$

- ① 通过键盘输入一个x的值。
- ② 采用关系表达式判断x的值，若 $x \geq 0$ ，计算 $y=5x+4$ 表达式，否则计算 $y=2x-3$ 表达式。
- ③ 输出y的值。实现流程图如右



4.3 if语句

if语句是根据给定的条件进行判断，以决定执行程序中某个分支程序段。if语句有3种使用形式：

单分支if语句:

```
if(条件表达式){  
    语句;  
}
```

双分支if语句:

```
if(条件表达式){  
    语句1;  
}else{  
    语句2;  
}
```

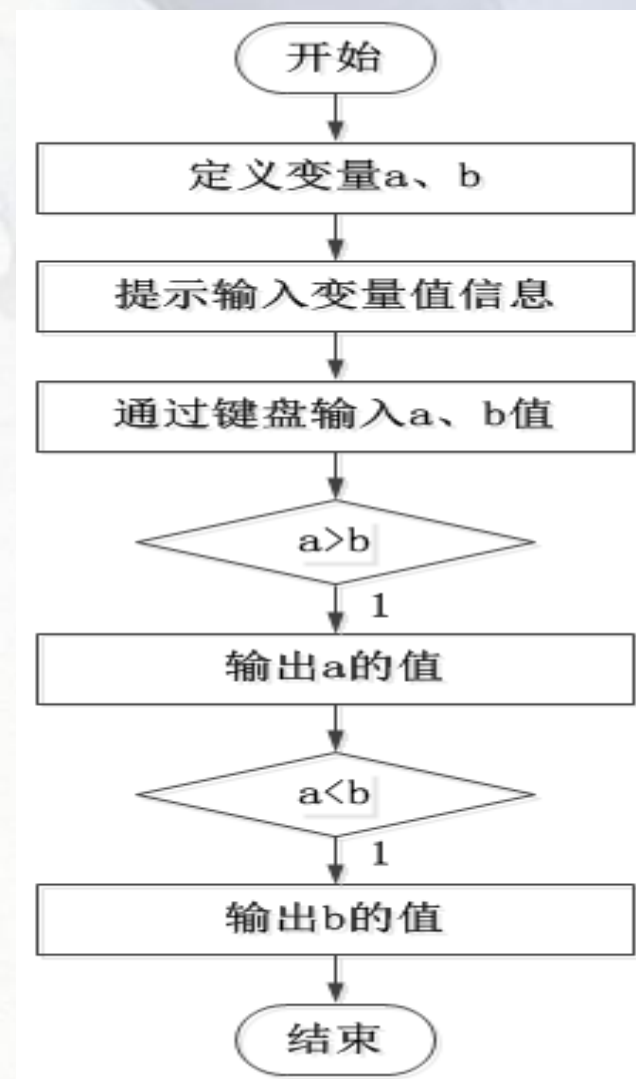
多分支if语句:

```
if(条件表达式){  
    语句1;  
}else if(条件表达式){  
    语句2;  
}  
.....  
else if(表达式n){  
    语句n;  
}else{  
    语句n+1;  
}
```

4.3.1 单分支if语句

【例题4.1】编写程序：输入任意的2个数，要求输出其中较大的数。

算法分析：定义两个实型变量a、b，采用关系表达式对a、b进行判断，若 $a > b$ 关系成立，则输出a，若 $a < b$ 关系成立，则输出b。
流程图如右图：



【例题4.2】编写程序：输入任意的3个数，要求按从小到大顺序输出。

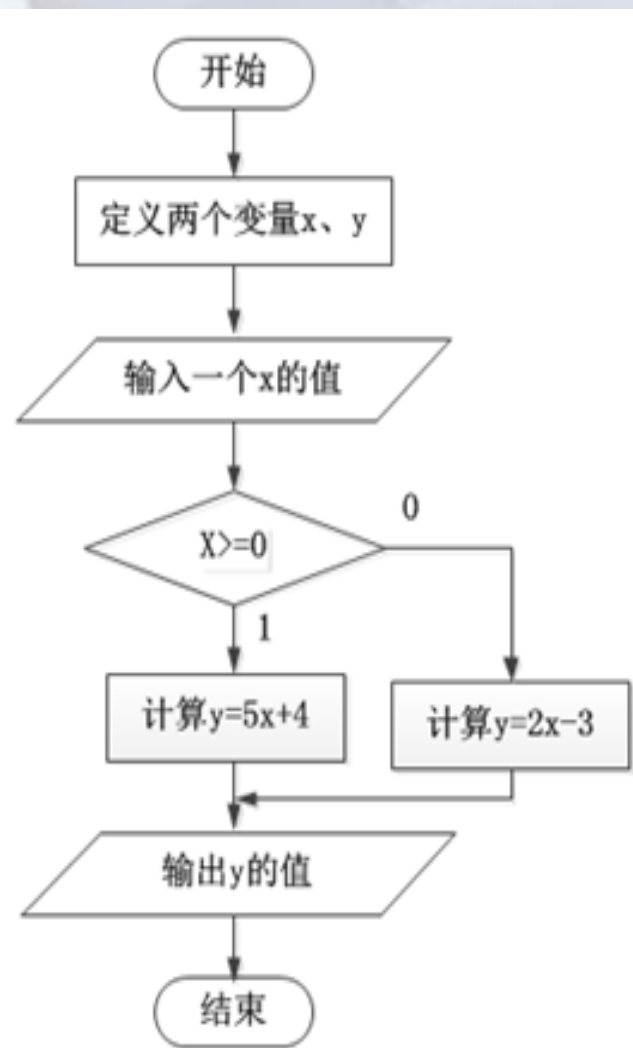
算法分析：定义三个实型变量a、b、c。用a与b进行比较，使a是较小的；用a与c进行比较，使a为3个数中较小的；用b与c进行比较，使b为剩下2个数中较小的，c就是3个数中较大的。最后按a、b、c的顺序输出。

另一算法：在3个数中找最大、最小数，3数之和减最大、最小得中间数。按最小数、中间数、最大数顺序输出。

【例题4.3】 计算分段函数的值，根据输入x的值输出函数y的值。

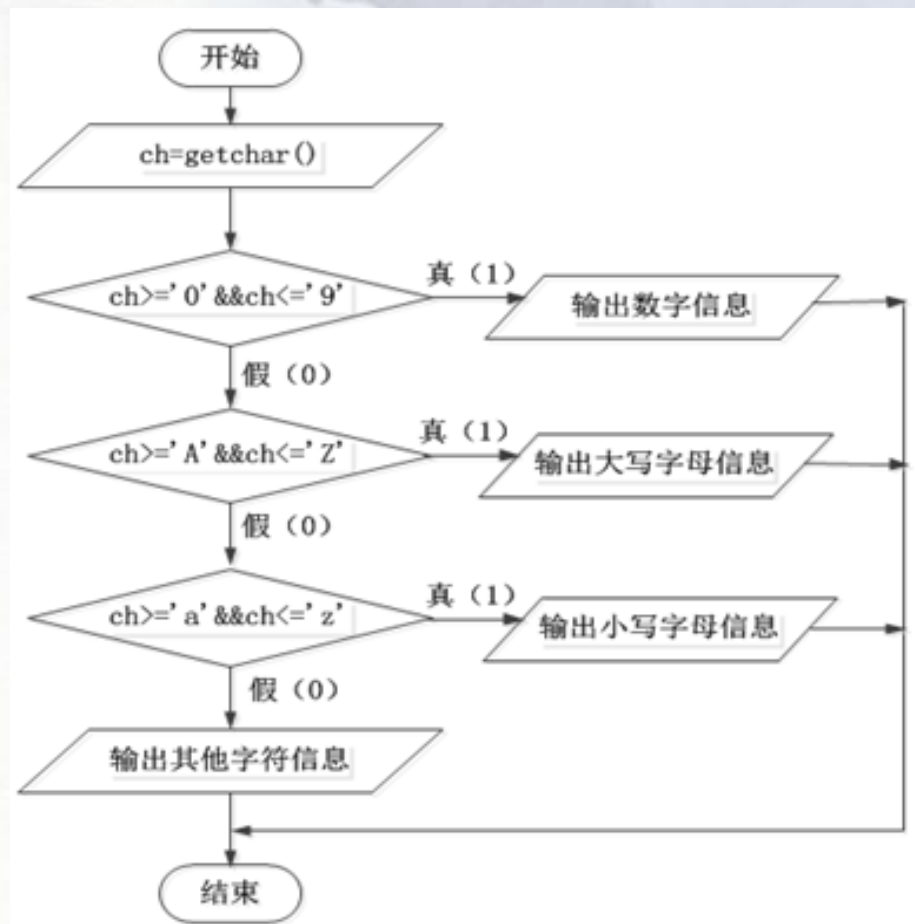
$$y = \begin{cases} 5x+4 & (x \geq 0) \\ 2x-3 & (x < 0) \end{cases}$$

算法分析： 定义变量x,y，输入一个x值，判断 $x \geq 0$ 若为真即表达值为1，则执行流 $5x+4$ 表达式，否则执行 $2x-3$ 表达式。流程图如右：



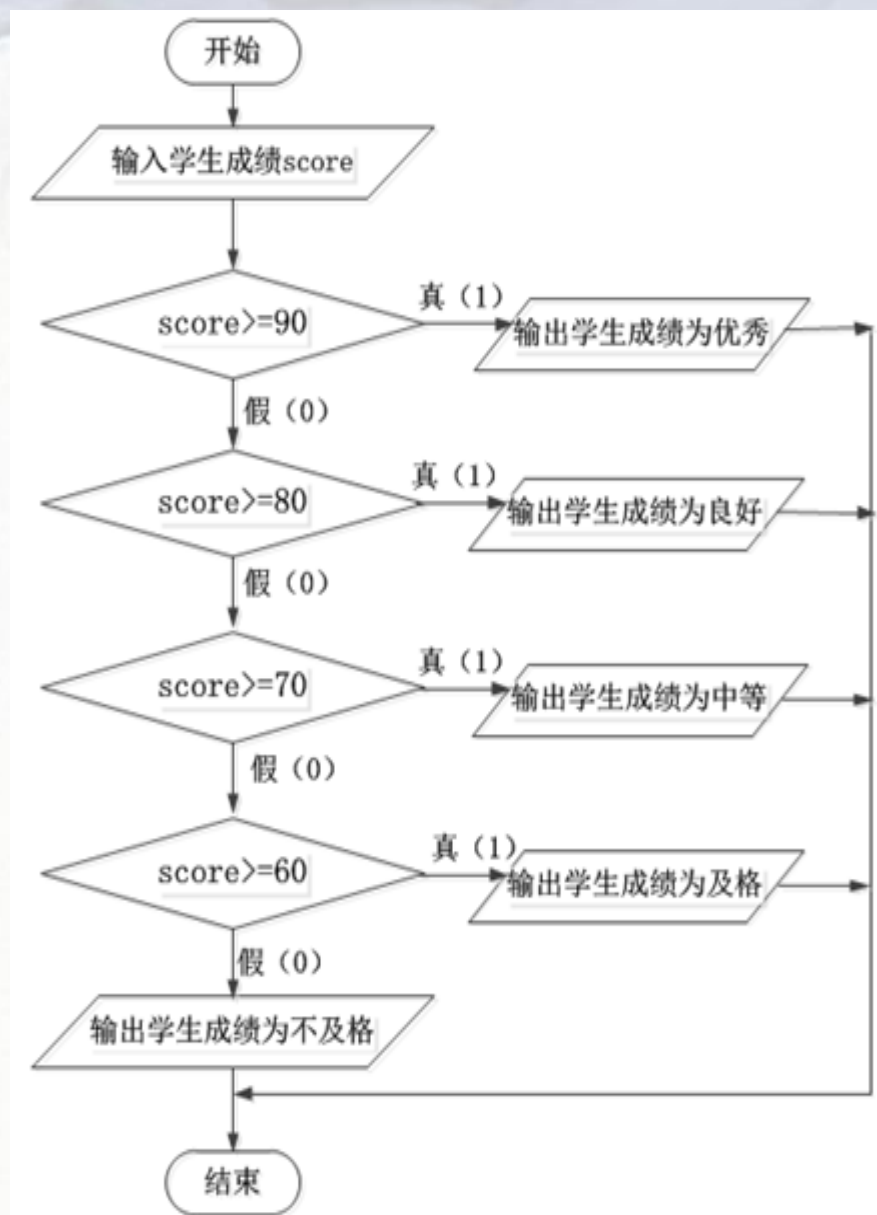
【例题4.4】编写程序：判断从键盘输入的字符是数字，是大写字母、是小写字母，还是其他字符，并输出相应的提示信息。

算法分析：定义一个字符型变量ch，调用字符输入函数getchar()实现输入字符。判断字符ch，若满足表达式 $ch \geq '0' \&\& ch \leq '9'$ ，则该字符为数字；否则继续判断，若满足表达式 $ch \geq 'a' \&\& ch \leq 'z'$ ，则该字符为小写字母；否则继续判断，若满足表达式 $ch \geq 'A' \&\& ch \leq 'Z'$ ，则该字符为大写字母；否则该字符为其它字符（如空格、回车等）。



【例题4.5】编写程序：输入一名学生的成绩，输出其对应的等级。90分以上为优秀，80分~89分为良好，70分~79分为中等，60分~69分为及格，60分以下为不及格。

算法分析：定义一个实型变量 `score`，调用 `scanf()` 输入函数实现成绩的输入。判断成绩 `score`，若满足 `score >= 90` 分，则输出成绩为优秀；否则继续判断成绩，若满足 `score >= 80` 分，则输出成绩为良好；否则继续判断成绩，若满足 `score >= 70` 分，则输出成绩为中等；否则继续判断成绩，若满足 `score >= 60` 分，则输出成绩为及格，否则输出成绩为不及格。



4.3.4 if语句的嵌套

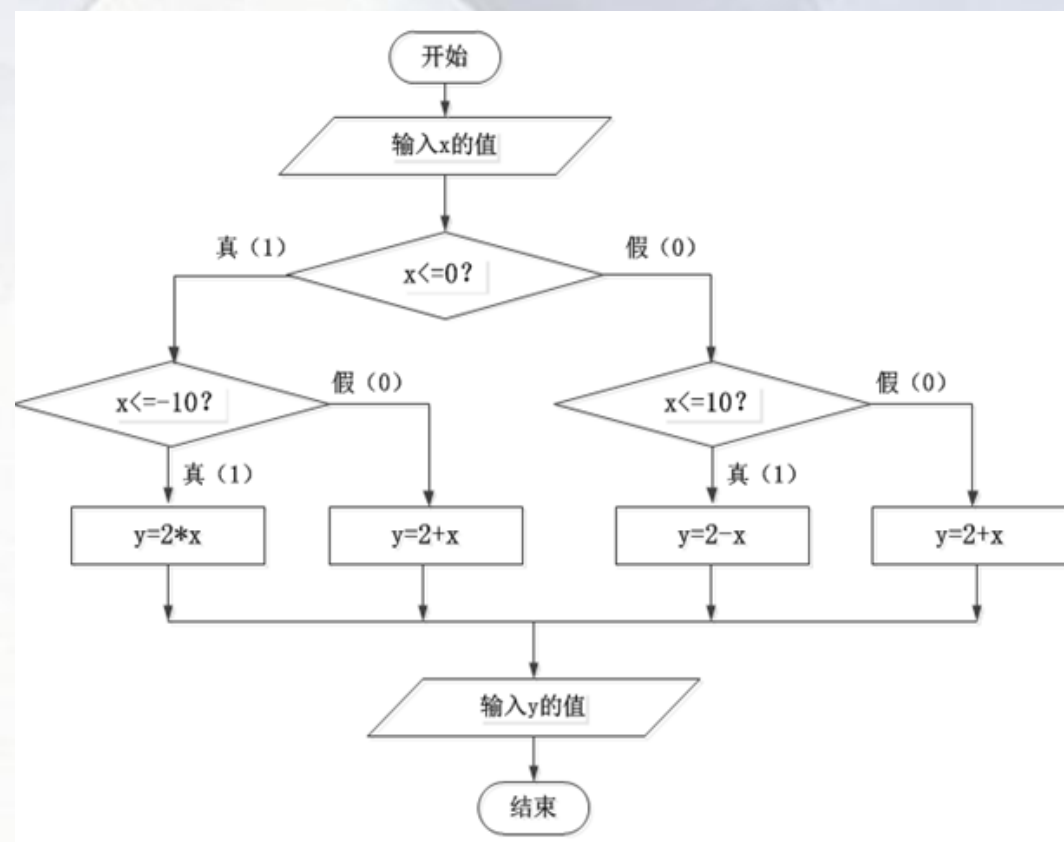
当if语句中的执行语句又是if语句时，称这种结构为if的嵌套结构。

```
if(表达式1){  
    if(表达式2){  
        语句1;  
    } else {  
        语句2;  
    }  
}  
else{  
  
}
```

【例题4.6】编写程序：运用if-else的嵌套形式求分段函数的值。

$$y = \begin{cases} 2*x & x \leq -10 \\ 2+x & -10 < x \leq 0 \\ x-2 & 0 < x \leq 10 \\ x/10 & x > 10 \end{cases}$$

算法分析：采用内嵌套形式编写。当 $x \leq 0$ ，是再判断是否大于-10，若是则计算 $2+x$ ；若不是则计算 $2*x$ 。当 $x \leq 0$ ，不是时，则再判断是否大于10，若是则计算 $x/10$ ；若不是则计算 $x-2$ 。最后输出 y 。



4.4 switch语句

【例题4.7】从键盘输入一名学生成绩(0 ~ 100之间), 输出该学生成绩对应的等级。其中A代表90分以上, B代表80 ~ 89, C代表70 ~ 79, D代表60 ~ 69, E代表60分以下, 如果成绩不在0 ~ 100之间则输出错误信息。

算法分析:从键盘输入一名学生成绩score。首先用if语句判断成绩是否在0 ~ 100之间, 若不在则输出提示信息; 若在则对其进行等级的判断。

程序代码 (if语句实现)

```
#include "stdio.h"
int main()
{
    float score;
    printf("\n please input a score:");
    scanf("%f",&score);
    if(score>=0&&score<=100){
        if(score>=90) {
            printf("学生成绩为 A!");
        }else if(score>=80) {
            printf("学生成绩为 B!");
        }else if(score>=70) {
            printf("学生成绩为 C!");
        }else if(score>=60) {
            printf("学生成绩为 D!");
        }else
            printf("学生成绩为 E!");
    }
    }else{
        printf("\n The score is wrong!");
    }
    return 0;
}
```

程序代码（switch语句实现）

```
switch((int)score/10)
{
    case 10:
    case 9:
        printf("学生成绩为A!\n");
        break;
    case 8:
        printf("学生成绩为B!\n");
        break;
    case 7:
        printf("\学生成绩为C!\n");
        break;
    case 6:
        printf("学生成绩为D!\n");
        break;
    default :
        printf("学生成绩为E!\n");
}
```

4.4 switch语句

switch(表达式)

```
{  
    case 常量表达式1:语句1; break;  
    case 常量表达式2:语句2; break;  
    case 常量表达式3:语句3; break;  
    case 常量表达式n:语句n; break;  
    ...  
    default :语句n+1;  
}
```

功能：跳出
switch语句

switch语句的功能

计算表达式的值，并与常量表达式值逐个相比较，若相等则执行其后语句，且不再比较并执行该语句后所有语句；若不相等，则执行default后的语句。

4.5 程序设计举例

【例题4.8】通过键盘输入任意的三个整数，要求输出其中最大数和最小数。

算法分析：

假设输入的三个数分别是：a、b、c。首先将a和b进行比较并大的数装入max中，小数装入min中；然后再将c分别与max比较，若大则替换max；与min比较，若小则替换min最后输出max和min就是三数中最大数和最小数。

【例题4.9】从键盘输入一个数学四则运算表达式($a+b$, $a-b$, $a*b$, a/b), 要求计算出该表达式的值。

程序分析:

应该设计两个整型变量 a 和 b 及一个存放运算符的字符型变量 ch , 根据 ch 的值来进行相应的表达式运算。因要实现 $+$ 、 $-$ 、 $*$ 、 $/$ 四种运算, 可以用 if 语句来实现, 也可用开关语句 $switch$ 语句来实现。即判断的表达式是运算符 ch 的值。最后输出表达式的运算结果。

本章小结

- 掌握关系运算符和关系表达式
- 掌握逻辑运算符和逻辑表达式
- 掌握if语句的三种格式的基本原理和基本用法
- 掌握switch语句的基本原理及基本用法